

BUNDESREPUBLIK DEUTSCHLAND

17 JUN 2005

PRIORITY  
DOCUMENTSUBMITTED OR TRANSMITTED IN  
COMPLIANCE WITH RULE 17.1(a) OR (b)

REC'D 26 FEB 2004

WIEG DOF

Prioritätsbescheinigung über die Einreichung  
einer Patentanmeldung

**Aktenzeichen:** 102 60 103.8

**Anmeldetag:** 19. Dezember 2002

**Anmelder/Inhaber:** ROBERT BOSCH GMBH, 70442 Stuttgart/DE

**Bezeichnung:** Verfahren und Vorrichtung zur Änderung von  
Software in einem Steuergerät sowie  
entsprechendes Steuergerät

**IPC:** G 11 C 16/06

Die angehefteten Stücke sind eine richtige und genaue Wiedergabe der  
ursprünglichen Unterlagen dieser Patentanmeldung.

München, den 14. Januar 2004  
Deutsches Patent- und Markenamt

Der Präsident

Im Auftrag

Wallner

19.12.02 Sy/Ho

ROBERT BOSCH GMBH, 70442 Stuttgart

Verfahren und Vorrichtung zur Änderung von Software in einem Steuergerät sowie  
entsprechendes Steuergerät

Stand der Technik

Die Erfindung betrifft ein Verfahren und eine Vorrichtung zur Änderung von Software in einem ersten Speicherbereich in einem Steuergerät sowie ein entsprechendes Steuergerät und ein Computerprogramm zur Ausführung des Verfahrens gemäß den unabhängigen Ansprüchen.

Insbesondere in der Automobilindustrie werden immer komplexere Funktionen mit Hilfe von Software in den einzelnen Steuergeräten umgesetzt. Immer häufiger ist es nötig, Änderungen der Software kurz vor, oder auch nach der Auslieferung an den Endkunden in das Steuergerät einfließen zu lassen. Die Änderungen der Software eines Steuergerätes werden im Allgemeinen dadurch behoben, dass man die Änderungen in einen neuen Software-Integrationsstand einfließen lässt und diese neu erstellte Software komplett in das Steuergerät lädt, insbesondere durch Speichern in einem sogenannten Flash-EPROM-Speicher, d. h. durch Flashen einbringt. Durch den vorhandenen Umfang der Software dauert das Flashen eines Steuergerätes beispielsweise schon jetzt bis zu 30 Minuten. Durch den starken Anstieg von Funktionen und Funktionalität und damit durch noch größeren Softwareumfang und eine größere Anzahl an Softwarepaketen wird diese Zeit in Zukunft noch stärker ansteigen.

Daher ist es Aufgabe der Erfindung, diesen Zeitaufwand zum Flashen eines Steuergerätes zu vermindern, wodurch sich auch noch weitere Vorteile ergeben.

### Vorteile der Erfindung

Die Erfindung geht aus von einem Verfahren und einer Vorrichtung zur Änderung von Software in einem ersten Speicherbereich in einem Steuergerät zur Steuerung von Betriebsabläufen, insbesondere bei einem Fahrzeug, sowie einem entsprechenden Steuergerät, wobei die Ausführung alter Softwareteile durch die Ausführung neuer Softwareteile ersetzt wird und die alten Softwareteile in dem ersten Speicherbereich eingeschrieben sind. In den meisten Fällen sind die Softwareänderungen, insbesondere Änderungen des Programmcodes klein gegenüber dem Anteil der Software, der nicht verändert werden muss. Vorteilhafter Weise werden nun gezielt die zu ändernden Anteile oder generell nur Teile einer Software in das Steuergerät geladen, wobei diese Softwareteile in der Beschreibung auch Patch-Container genannt werden bzw. in einem solchen Patch-Container enthalten sind.

Zweckmäßiger Weise werden also nur die neuen Softwareteile in einen zweiten Speicherbereich eingeschrieben, wobei durch eine erste Verzweigung in dem ersten Speicherbereich statt der alten Softwareteile im ersten Speicherbereich die neuen Softwareteile im zweiten Speicherbereich ausgeführt werden und nach Ausführung der neuen Softwareteile durch eine zweite Verzweigung im zweiten Speicherbereich wieder in den ersten Speicherbereich rückverzweigt wird, wobei die Ausführung der weiteren von den alten Softwareteilen verschiedenen Software im ersten Speicherbereich fortgesetzt wird und die alten Softwareteile im ersten Speicherbereich verbleiben.

D. h. es wird, um die Änderungen der bestehenden Software, also der alten Softwareteile zu realisieren, eine insbesondere zentrale Patchverwaltung installiert, mit welcher die zu ändernden Teile der Software gezielt durch Einsprung und Aussprung geändert werden können. Dadurch kann vorteilhafter Weise zum Einen der hohe Zeitaufwand beim Flashen großer Softwareabschnitte oder Softwareteile deutlich verringert werden, da zum Einen nicht, wie bereits oben angeklungen, die gesamte Software neu eingeschrieben wird und zum Anderen kann auch der Zeitaufwand eingespart werden, der nötig ist zur Änderung der Software, also insbesondere diese zu löschen und neu einzuschreiben.

Dadurch ergibt sich weiterhin eine höhere Flexibilität bei der Änderung von Software und Softwareteilen und zum Anderen eine geringere Fehleranfälligkeit durch möglichst geringes Überschreiben bzw. Löschen und neu Beschreiben von Softwareteilen.

5 Dies wird zweckmäßiger Weise insbesondere dadurch realisiert, dass der zweite Speicherbereich, also der Patch-Container-Bereich nur dazu verwendet wird, die neuen Softwareteile aufzunehmen und in den Programmlauf bzw. den Softwareablauf einzubinden.

10 Dabei kann zweckmäßiger Weise die erste Verzweigung und die zweite Verzweigung durch wenigstens eine verkettete Liste realisiert werden. Verkettete Listen stellen dabei eine platzsparende Möglichkeit zur Ablage von Daten, insbesondere von Softwareteilen zur Verfügung, welche zugleich eine zeitliche als auch eine logische Ordnung besitzen. Da auf diese Art und Weise von vergleichbaren oder gleichen Softwareteilen mehrere  
15 Versionen vorhanden sein können, sind verkettete Liste ein gut geeigneter Ansatz zur Ablage einer im Voraus nicht bestimmbar Anzahl von Versionen verschiedener Softwareteile. Bei verketteten Listen wird für jede Version eines Softwareteils bzw. für jeden Softwareteil ein Zeiger bzw. ein Verweis auf die nächste Version dieses Softwareteils geführt. Ein spezieller Zeigerzustand kann zur Codierung für den Fall  
20 verwendet werden, dass die zugehörige Version die aktuellste Version des jeweiligen Softwareteils ist. Ebenso kann dieser spezielle Zeigerzustand als Kennung bzw. Identifier für den jeweiligen Softwareteil dienen. Als Verweis werden dabei relative oder absolute Adressen verwendet. Bei einer verketteten Liste wird somit zusammen mit einem jeweils abgelegten Softwareteil bzw. zusammen mit einer Informationseinheit wie eines  
25 Datensatzes der den Softwareteil enthält ein Verweis auf die Ablagestelle der jeweils nächsten logisch verknüpften Informationseinheit gespeichert, insbesondere hier entweder des nächsten neuen Softwareteils oder Datensatzes mit dem neuen Softwareteil oder auch der weiteren dem alten Softwareteil folgenden Software. So können die Ablagestellen auch erst zu dem Zeitpunkt belegt werden, zu dem die zu speichernde bzw.  
30 neu auszuführende Information zur Verfügung steht. So ist es außerdem möglich, dieser Information, die jeweils nächste, freie Ablagestelle zuzuweisen, wodurch dieses Vorgehen außerdem zu einer lückenlosen Nutzung des Speichers führt.

Zweckmäßiger Weise wird nun als erste Verzweigung, quasi zum Aussprung aus dem ersten Speicherbereich, eine Anfangsadresse der neuen Softwareteile verwendet, wobei mit dieser wenigstens teilweise die alten Softwareteile, also die zu ändernden Softwareteile überschrieben werden können. Alternativ kann auch Speicherplatz zur  
5 Einfügung der Aussprungadresse im ersten Speicherbereich vorgesehen sein. Es wird somit im ersten Speicherbereich ein Aussprung in den oder zum Patch-Container gesetzt. Als zweite Verzweigung, quasi als Rücksprung wird dann vorteilhafter Weise eine Anfangsadresse der weiteren von den alten Softwareteilen verschiedenen Software verwendet, so dass lückenlos nach dem Umspringen der alten Softwareteile die weitere,  
10 nachfolgende Software im ersten Speicherbereich verwendet bzw. abgearbeitet wird.

Vorteilhafter Weise sind in den neuen Softwareteilen Informationen enthalten, die angeben, welche alten Softwareteile ersetzt werden sollen und/oder solche Informationen, die zeigen, durch welche neuen Softwareteile die alten Softwareteile ersetzt werden  
15 sollen. Zweckmäßiger Weise enthält dazu der zweite Speicherbereich, also insbesondere jeder eingelesene Patch-Container in einer Patch-Tabelle, die die neuen Softwareteile sowie unter Umständen Zusatzinformation enthält, neben wenigstens einem neuen Softwareteil eine Adresse für die erste Verzweigung, eine Adresse für die zweite Verzweigung und eine Adresse für den Beginn des alten Softwareteils, das durch das  
20 wenigstens eine neue Softwareteil zu ersetzen ist. Als zusätzliche Information kann vorteilhafter Weise auch die Länge des wenigstens einen neuen Softwareteils und/oder auch des wenigstens einen alten Softwareteils im Patch-Container enthalten sein. Diese Elemente, also eine Adresse für die erste Verzweigung, eine Adresse für die zweite Verzweigung, eine Adresse für den Beginn des alten Softwareteils sowie unter  
25 Umständen die Länge des neuen Softwareteils und/oder des alten Softwareteils können zu einem Datensatz im Patch-Container, also in dem zweiten Speicherbereich, insbesondere der Patch-Tabelle, zusammengefasst sein. Zweckmäßiger Weise ist so die Information, die nötig ist, um einen alten Softwareteil zu ersetzen, für jeden alten Softwareteil jeweils in einem solchen Datensatz zusammengefasst.

30 Dabei kann vorteilhafter Weise der erste Speicherbereich als eine erste Tabelle und der zweite Speicherbereich als eine zweite Tabelle (Patch-Tabelle) im selben Speicher realisiert werden.

Dabei kann außerdem der erste Speicherbereich und der zweite Speicherbereich zweckmäßiger Weise in gleich große Softwareabschnitte aufgeteilt sein, wobei in jeden Softwareabschnitt des zweiten Speicherbereichs ein neuer Softwareteil einschreibbar ist.

5      Zweckmäßiger Weise kann dann auch jeder Datensatz oder jeder Softwareabschnitt entsprechend der Ausführungsform mit einer Kennung versehen werden, die eine Zuordnung eines alten Softwareteils und eines diesen ersetzenden neuen Softwareteils gestattet.

10      Weitere Vorteile und vorteilhafte Ausgestaltungen ergeben sich aus der Beschreibung sowie den Merkmalen der Ansprüche.

Zeichnung

15      Die Erfindung wird im Weiteren anhand der in der Zeichnung dargestellten Figuren näher erläutert. Dabei zeigt:

Figur 1 ein Steuergerät mit entsprechender Software, bei dem alte Softwareteile durch neue Softwareteile ersetzt werden und

20      Figur 2 den ersten Speicherbereich mit der Steuergerätesoftware sowie den erfindungsgemäßen Patch-Container und

25      Figur 3 beispielhaft einen möglichen Datensatz im Patch-Container.

Beschreibung der Ausführungsbeispiele

Figur 1 zeigt ein Steuergerät 100 mit einer Verarbeitungseinheit 101, insbesondere einem Mikrocontroller sowie Speichermittel 102, insbesondere aufgeteilt in zwei  
30      Speicherbereiche 103 und 104. Diese Speicherbereiche 103 und 104 können sich im selben Speicher oder in verschiedenen Speichern des Steuergeräts 100 befinden. Über eine Schnittstelle 105, die neben einer leitungsgebundenen auch eine leitungslose Verbindung repräsentieren kann, werden die entsprechenden neuen Softwareteile von einer Quelle 106, beispielsweise einem weiteren Rechner, mit zweitem Speichermittel

107 in das Steuergerät 100 eingebracht. Dadurch, dass nur die zu ändernden Daten übermittelt werden, also nur die neuen Softwareteile und nicht die gesamte Software, was deutlich geringere Übertragungsraten zur Folge hat, können insbesondere auch Luftschnittstellen, also Funk, Ultraschall, Infrarot usw. Einsatz finden. Daneben ist an dieser Stelle aber die leitungsgebundene Übertragung ebenso verwendbar.

Figur 2 zeigt nun einen ersten Speicherbereich 200 und einen zweiten Speicherbereich 201, insbesondere in Form einer Tabelle dargestellt. Der erste Speicherbereich enthält dabei Zellen bzw. Softwareabschnitte 205 bis 216. Die entsprechenden Adressen der Zellen 205 bis 216 sind in Block 203 jeweils abgelegt. Ebenso enthält ein zweiter Speicherbereich 201, also die Patch-Tabelle, mehrere Zellen 217 bis 223, ebenfalls Softwareabschnitte, in die die sogenannten Patch-Container eingespeichert werden können. Auch hier sind entsprechende Adressen im Block 204 zur jeweiligen Zelle, also zum jeweiligen Patch-Container abgelegt.

Wie in Figur 2 zu erkennen, wird in den alten Softwareteilen aus dem ersten Speicherbereich 200 ein Aussprung nach dem Softwareabschnitt 207 im Softwareabschnitt 208 auf die Patch-Tabelle 201 generiert. Dabei wird der alte Softwareteil 208 wenigstens teilweise mit der Sprungadresse, also der Verzweigung zu Softwareabschnitt 218 überschrieben. D. h. die erste Änderung A1 beginnt mit einer Verzweigung V1A1 aus dem ersten Speicherbereich 200 in den zweiten Speicherbereich 201. Nach Ablauf der Softwareabschnitte 218, 219 wird aus Softwareabschnitt 220 mit einer zweiten Verzweigung V2A1 wieder in den ersten Speicherbereich, den Softwareabschnitt 210 rückverzweigt bzw. zurückgesprungen. So können also die alten Softwareteile in den Softwareabschnitten 208 und 209 durch die neuen Softwareteile in den Abschnitten 218 und 219 ersetzt, also umsprungen werden. Dabei bleiben die alten Softwareteile in den Abschnitten 208 und 209 weiter im ersten Speicherbereich enthalten.

Eine zweite Änderung A2 zeigt beispielsweise einen Aussprung nach Abschnitt 212 zu Beginn des Abschnittes 223 im zweiten Speicherbereich 201 und einen Rücksprung zu Beginn des Speicherabschnitts 214. D. h. durch die erste Verzweigung V1A2 und den Rücksprung oder die zweite Verzweigung V2A2 wird der Softwareabschnitt 213 durch den Softwareabschnitt 223 bezüglich des Softwareablaufs ersetzt, wobei wiederum der

alte Softwareteil im Softwareabschnitt 213 zumindest teilweise erhalten bleibt. Lediglich der Teil wird ersetzt, der die Sprungadresse für die Patch-Tabelle enthält.

5 Um diesen Zustand, wie eben mit Figur 2 beschrieben, zu erzeugen, wird die Information benötigt, welche Softwareteile, also alte Softwareteile aus Speicherbereich 200 wie geändert werden sollen. Diese Information wird in die Patch-Tabelle 201 geladen.

10 Mit einer Patch-Verwaltung werden beim Neuanlauf des Steuergerätes die Adressbezüge zwischen Speicherbereich 200 und Speicherbereich 201 hergestellt. Die Patch-Verwaltung kann z. B. im Boot-Loader, also dem Startprogramm, das beim Anlauf des Steuergerätes abgearbeitet wird, untergebracht sein. Dabei kann je nach Aufwand erfindungsgemäß eine Vielzahl von Informationen im Rahmen der Patch-Verwaltung in der Patch-Tabelle enthalten sein. Diese Informationen, diese Elemente, können in einzelnen Datensätzen zusammengefasst werden, wie dies in Figur 3 dargestellt ist. Die  
15 Verwaltung kann auch von der Verarbeitungseinheit 101 des Steuergerätes selbst realisiert werden, indem die entsprechenden Informationen im Patch-Container hinterlegt sind.

20 Um die Datensätze für die Patch-Tabelle zu generieren, bietet sich dabei ein Hilfsmittel, insbesondere im Rechner 106 an, welches die Änderungen des Quellcodes analysiert und danach die Datensätze bzw. die Softwareteile für die Patch-Tabelle erzeugt. Diese Datensätze bestehen beispielsweise aus einer Aussprungadresse, aus der Originalsoftware, also den alten Softwareteilen, einer Zieladresse für die Patch-Tabelle, also einem Verweis auf einen Patch-Container, dem neuen Softwareteil, also  
25 beispielsweise geänderter oder neuer Hexadezimalcode, beispielsweise auch aus der Länge des geänderten Programmcodes, also der Länge des alten Softwareteils oder auch der Länge des neuen Programmcodes, also des neuen Softwareteils sowie aus der Rücksprungadresse in den ersten Speicherbereich, also zur weiteren vom alten Softwareteil verschiedenen Software also insbesondere ins ursprüngliche Programm.  
30 Weiterhin kann einem Patch-Container ein Identifier oder eine Kennung zugeordnet und damit im Datensatz enthalten sein.

Dabei sind nicht alle die genannten Informationen zur Realisierung des erfindungsgemäßen Verfahrens notwendig, so kann z.B. aus Anfangsadresse und Länge



des Softwareteils die Endadresse bestimmt werden. Dennoch kann die Verwendung der Information entweder einen Zeitvorteil bedeuten, da eine Adresse nicht berechnet werden muss, sondern angegeben ist oder kann redundante Information in diesem Zusammenhang eine Überprüfbarkeit der Softwareänderung und damit eine Erhöhung der Sicherheit ermöglichen.

Gleichermaßen kann mit einer solch aufgebauten Patch-Verwaltung auch eine Änderung in der Software wieder rückgängig gemacht werden. Bei einem solchen Rückgängigmachen könnte dann Speicherplatz im ersten Speicherbereich für die Aussprunadressen vorgesehen werden, so dass bei einer Wiederverwendung des alten Softwareteils dieses nicht teilweise überschrieben ist.

Als Beispiel eines solchen Patch-Containers oder Datensatzes ist in Figur 3 Block 300 dargestellt. Dabei ist mit 301 beispielsweise eine Programmadresse, also die Adresse des alten zu ersetzenden Softwareteils angegeben, mit Block 302 die Adresse in der Patch-Tabelle. Block 303 gibt den neuen insbesondere Programmcode, also das neue Softwareteil an, welches das alte Softwareteil ersetzen soll. Mit Block 304 ist beispielsweise die Länge des neuen Softwareteils, also die Codelänge in Bit, Byte oder einer beliebigen anderen Menge angegeben, und schließlich zeigt Block 305 die Rücksprunadressen zu den weiteren Softwareteilen, also insbesondere zum Originalprogramm in den ersten Speicherbereich 200.

Daneben kann, wie schon erwähnt, eine Kennung, also ein Identifier ebenso enthalten sein, beispielsweise anstatt der Codelängenangabe oder auch zusätzlich. Ein solchen Patch-Container wie Block 300 bzw. die gesamte Patch-Tabelle kann sehr einfach über ein Bussystem in das Steuergerät geladen werden, z. B. über ein CAN-Bus-System mit den CAN-Botschaften. Wird eine Kennung oder ein Identifier wie angesprochen verwendet, also einem Patch-Container zugeordnet, kann man die Änderungen, z. B. Fehlermeldungen von Kunden, ganz gezielt ändern, nachverfolgen und auch eine Historie der Änderungen aufbauen und auch speichern, ohne die Transparenz aus dem stabilen ausgetesteten Programmcode, also die fehlerfreie Software zu verlieren. Zeigt eine Korrektur nicht die gewünschte Reaktion, kann man leicht wieder den alten Zustand herstellen, also den Patch-Container einfach herausnehmen. Durch die geringe Datenrate,

die das erfindungsgemäße Verfahren bedient, ist eine einfache Korrektur oder Erweiterung der Software auch über die Luftschnittstelle, wie bereits erwähnt, möglich.

5 Weiterhin vorteilhaft ist es, den Speicherbereich für die Daten, also die Software, in gleich große Datenfiles, also gleich große Softwareabschnitte, zu segmentieren, die dann ebenfalls, wie beim Verfahren, wie oben beschrieben, miteinander verkettet und für die Veränderungen in der Patch-Tabelle abgelegt werden. D. h. die Zellen oder Datenpakete bzw. die darin abgelegte Software im ersten Speicherbereich 200 und im zweiten Speicherbereich 201 müssen nicht zwangsläufig gleich groß sein, sondern es wird über  
10 die Adressen und die Patch-Verwaltung gemanagt. Es kann aber vorteilhaft sein, gleich große Softwareabschnitte für den ersten und zweiten Speicherbereich zu verwenden und immer solche kompletten Softwareabschnitte als Datensatz auszutauschen bzw. die alten Softwareteile durch die neuen zu ersetzen.

15 Durch die dargestellte Erfindung in all ihren Ausprägungen erhält man somit ein inkrementelles Flashen eines Steuergerätes mit all den genannten Vorteilen.

19.12.02 Sy/Ho

5

ROBERT BOSCH GMBH, 70442 Stuttgart

10

Ansprüche

15

20

25

30

1. Verfahren zur Änderung von Software in einem ersten Speicherbereich in einem Steuergerät zur Steuerung von Betriebsabläufen, wobei die Ausführung alter Softwareteile durch die Ausführung neuer Softwareteile ersetzt wird und die alten Softwareteile in dem ersten Speicherbereich eingeschrieben sind, dadurch gekennzeichnet, dass die neuen Softwareteile in einem zweiten Speicherbereich eingeschrieben werden und durch eine erste Verzweigung im ersten Speicherbereich statt der alten Softwareteile im ersten Speicherbereich, die neuen Softwareteile im zweiten Speicherbereich ausgeführt werden, wobei nach Ausführung der neuen Softwareteile durch eine zweite Verzweigung im zweiten Speicherbereich, wieder in den ersten Speicherbereich rückverzweigt wird und die Ausführung der weiteren von den alten Softwareteilen verschiedene Software im ersten Speicherbereich fortgesetzt wird, wobei die alten Softwareteile im ersten Speicherbereich verbleiben.
2. Verfahren nach Anspruch 1, dadurch gekennzeichnet, dass der zweite Speicherbereich nur dazu verwendet wird die neuen Softwareteile aufzunehmen.
3. Verfahren Anspruch 1, dadurch gekennzeichnet, dass die erste Verzweigung und die zweite Verzweigung durch wenigstens eine verkettete Liste realisiert wird.
4. Verfahren nach Anspruch 1, dadurch gekennzeichnet, dass als erste Verzweigung eine Anfangsadresse der neuen Softwareteile verwendet wird, wobei mit dieser wenigstens teilweise die alten Softwareteile überschrieben werden.

5. Verfahren nach Anspruch 1, dadurch gekennzeichnet, dass als zweite Verzweigung eine Anfangsadresse der weiteren von den alten Softwareteilen verschiedene Software verwendet wird.

5 6. Verfahren nach Anspruch 1, dadurch gekennzeichnet, dass in den neuen Softwareteilen Informationen enthalten sind, die angeben welche alten Softwareteile ersetzt werden sollen.

10 7. Verfahren nach Anspruch 1, dadurch gekennzeichnet, dass in den neuen Softwareteilen Informationen enthalten sind, die angeben durch welche neuen Softwareteile die alten Softwareteile ersetzt werden sollen.

15 8. Verfahren nach Anspruch 1, dadurch gekennzeichnet, dass der zweite Speicherbereich neben wenigstens einem neuen Softwareteil eine Adresse für die erste Verzweigung, eine Adresse für die zweite Verzweigung und eine Adresse für den Beginn des alten Softwareteils, das durch das wenigstens eine neue Softwareteil zu ersetzen ist enthält.

20 9. Verfahren nach Anspruch 8, dadurch gekennzeichnet, dass der zweite Speicherbereich weiterhin die Länge des wenigstens einen neuen Softwareteils und/oder des wenigstens einen alten Softwareteils enthält.

25 10. Verfahren nach Anspruch 8 oder 9, dadurch gekennzeichnet, dass die Elemente des Anspruchs 8 und/oder die Elemente des Anspruchs 9 zu einem Datensatz im zweiten Speicherbereich zusammengefasst sind.

30 11. Verfahren nach Anspruch 10, dadurch gekennzeichnet, dass wenigstens zwei alte Softwareteile und wenigstens zwei neue Softwareteile, die diese ersetzen vorgesehen sind wobei die entsprechenden Elemente des Anspruchs 8 und/oder 9 zu jeweils einem Datensatz zusammengefasst und im zweiten Speicherbereich eingeschrieben sind.

12. Verfahren nach Anspruch 1, dadurch gekennzeichnet, dass als erster Speicherbereich eine erste Tabelle und als zweiter Speicherbereich eine zweite Tabelle im selben Speicher vorgesehen sind.

13. Verfahren nach Anspruch 1 oder 12, dadurch gekennzeichnet, dass der erste Speicherbereich und der zweite Speicherbereich in gleichgroße Softwareabschnitte aufgeteilt ist, wobei in jeden Softwareabschnitt des zweiten Speicherbereichs ein neuer Softwareteil einschreibbar ist.

14. Verfahren nach Anspruch 10 oder 13, dadurch gekennzeichnet, dass jeder Datensatz oder jeder Softwareabschnitt mit einer Kennung versehen ist.

15. Verfahren nach Anspruch 14, dadurch gekennzeichnet, dass die Kennung für einen Softwareabschnitt im ersten Speicherbereich, der einen alten Softwareteil enthält und die Kennung für den entsprechenden Softwareabschnitt mit dem neuen Softwareteil, der den alten Softwareteil ersetzt gleich ist.

16. Vorrichtung zur Änderung von Software in einem ersten Speicherbereich in einem Steuergerät zur Steuerung von Betriebsabläufen, wobei die Ausführung alter Softwareteile durch die Ausführung neuer Softwareteile ersetzt wird und die alten Softwareteile in dem ersten Speicherbereich eingeschrieben sind, dadurch gekennzeichnet, dass Mittel enthalten sind, die die neuen Softwareteile in einen zweiten Speicherbereich und eine erste Verzweigung in den ersten Speicherbereich einschreiben, wodurch statt der alten Softwareteile im ersten Speicherbereich, die neuen Softwareteile im zweiten Speicherbereich ausgeführt werden, wobei die Mittel auch eine zweite Verzweigung in den zweiten Speicherbereich einschreiben, wodurch nach Ausführung der neuen Softwareteile, wieder in den ersten Speicherbereich rückverzweigt wird und die Ausführung der weiteren von den alten Softwareteilen verschiedenen Software im ersten Speicherbereich fortgesetzt wird, wobei die alten Softwareteile im ersten Speicherbereich verbleiben.

17. Steuergerät mit einem ersten Speicherbereich, in dem alte Softwareteile und weitere von den alten Softwareteilen unterschiedliche Softwareteile gespeichert sind, dadurch gekennzeichnet, dass das Steuergerät einen zweiten Speicherbereich enthält, welcher neue, die alten Softwareteile ersetzende Softwareteile enthält, dadurch gekennzeichnet, dass Mittel enthalten sind, die die neuen Softwareteile in einen zweiten Speicherbereich und eine erste Verzweigung in den ersten Speicherbereich einschreiben, wodurch statt der alten Softwareteile im ersten Speicherbereich, die neuen Softwareteile im zweiten Speicherbereich ausgeführt werden, wobei die Mittel auch eine zweite Verzweigung in den zweiten

Speicherbereich einschreiben, wodurch nach Ausführung der neuen Softwareteile, wieder in den ersten Speicherbereich rückverzweigt wird und die Ausführung der weiteren von den alten Softwareteilen verschiedenen Software' im ersten Speicherbereich fortgesetzt wird, wobei die alten Softwareteile im ersten Speicherbereich verbleiben.

5

18. Computerprogramm mit Programmcode zur Durchführung aller Schritte nach einem der Ansprüche 1 bis 15, wenn das Programm auf einem Computer, insbesondere in einem Steuergerät ausgeführt wird.

19.12.02 Sy/Ho

5

ROBERT BOSCH GMBH, 70442 Stuttgart

10

Verfahren und Vorrichtung zur Änderung von Software in einem Steuergerät sowie  
entsprechendes Steuergerät

Zusammenfassung

15

Verfahren und Vorrichtung zur Änderung von Software in einem ersten Speicherbereich  
in einem Steuergerät zur Steuerung von Betriebsabläufen, wobei die Ausführung alter  
Softwareteile durch die Ausführung neuer Softwareteile ersetzt wird und die alten  
Softwareteile in dem ersten Speicherbereich eingeschrieben sind, wobei die neuen  
Softwareteile in einem zweiten Speicherbereich eingeschrieben werden und durch eine  
erste Verzweigung im ersten Speicherbereich statt der alten Softwareteile im ersten  
Speicherbereich, die neuen Softwareteile im zweiten Speicherbereich ausgeführt werden,  
wobei nach Ausführung der neuen Softwareteile durch eine zweite Verzweigung im  
zweiten Speicherbereich, wieder in den ersten Speicherbereich rückverzweigt wird und  
die Ausführung der weiteren von den alten Softwareteilen verschiedene Software im  
ersten Speicherbereich fortgesetzt wird, wobei die alten Softwareteile im ersten  
Speicherbereich verbleiben.

20

25

(Figur 2)

112

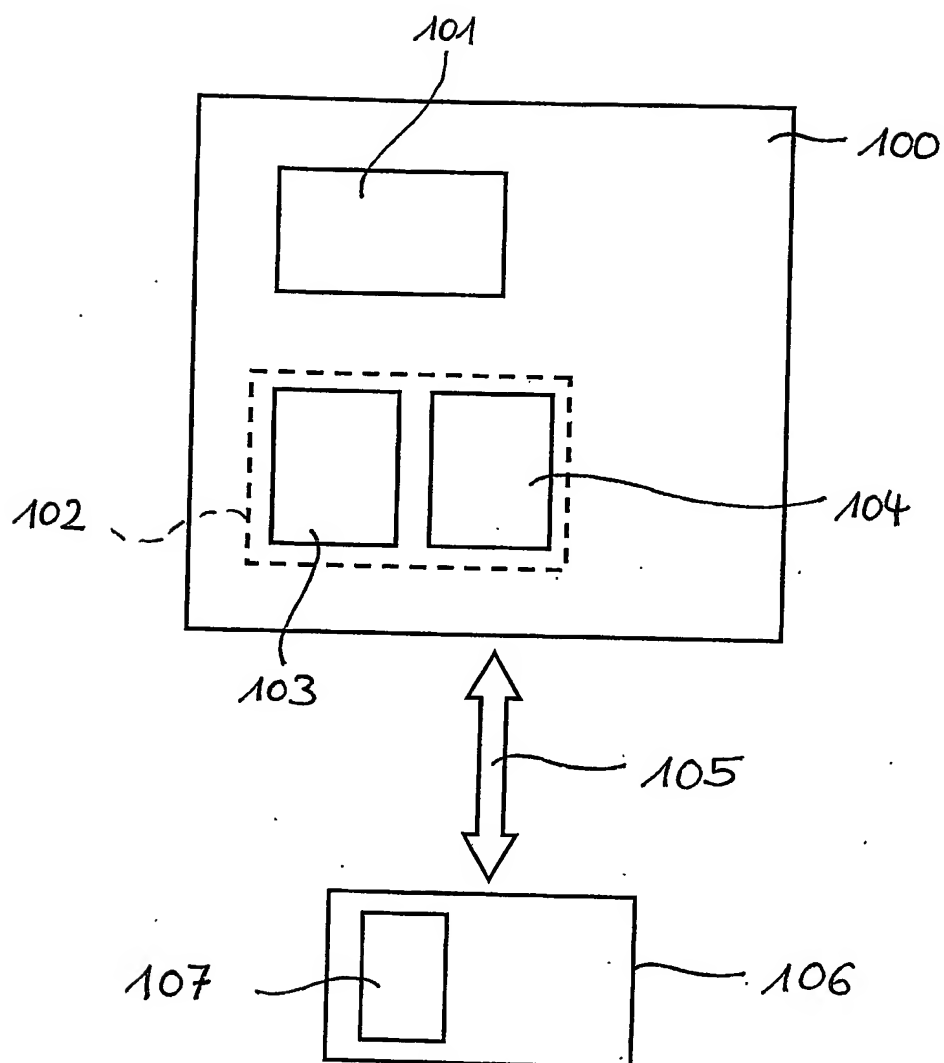


Fig. 1



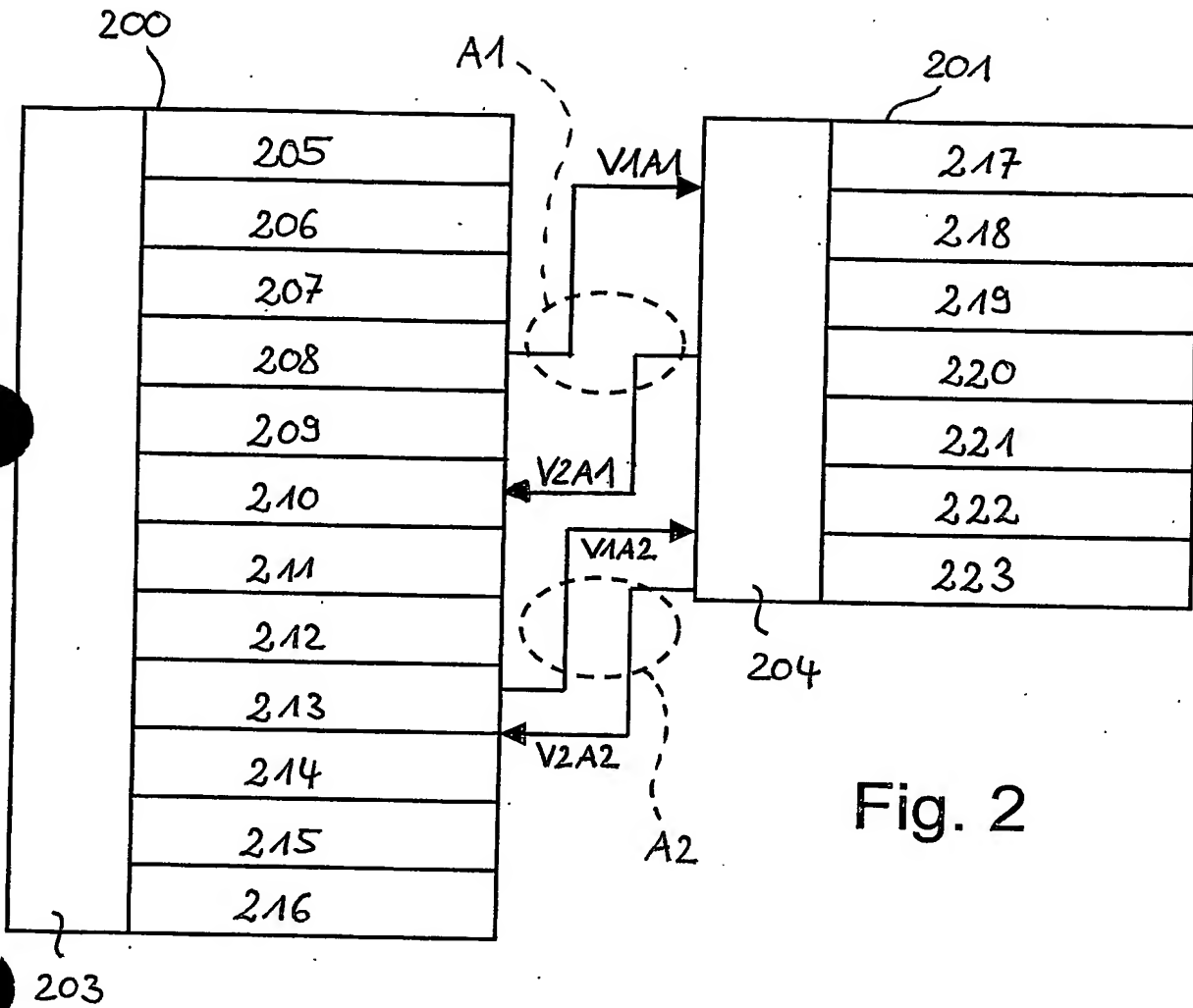


Fig. 2

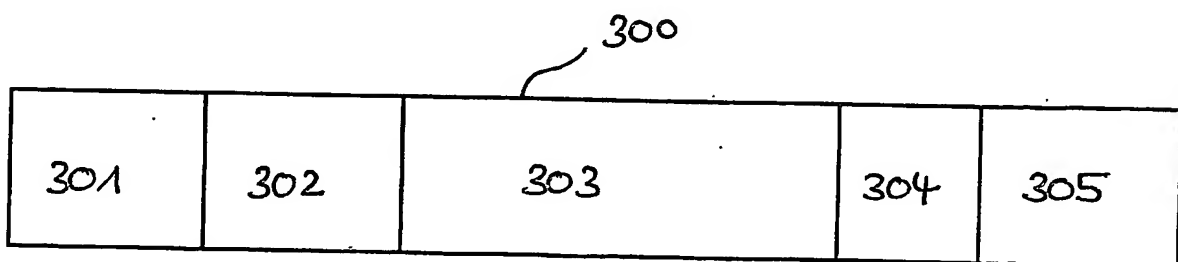


Fig. 3